Survey on Design of Quantum Computing Analysis Techniques

Milan Jain

ABSTRACT- As it is a modern era of science and technology and the humankind is to elaborate their thinking in the field of medicine and different technology sector. Quantum computing is essentially harnessing and exploiting the amazing laws of quantum mechanics to process information. A quantum computer uses quantum bits or qubits. Qubit is a quantum system that encodes the zero and the one into two distinguishable quantum states, qubits behave quantumly. Quantum computing technique is discussed about the phenomena of superposition and entanglement. Superposition is one particle property while entanglement is a characteristic of two or more particles. If use our classical algorithms on a quantum computer, it perform the operation as the way done by classical computer. To show the superiority of quantum it needs to use new algorithms which can exploit the phenomenon of quantum parallelism. In this paper the comparative analysis is discuss according to various quantum parameters.

Index Terms— Black box quantum computing known as oracle, Hadamard transform, Hadamard gate, Quantum algorithm, Superposition, Eigen value, Eigen state.

1 INTRODUCTION

A quantum algorithm is sequence of operations that makes the computer do a specific task, like solving a certain problem. You have all learned how to write algorithms and to implement them in different high-level languages like Mat lab, Mathematic, C, Fortran etc. Quantum Computing is still on a much more basic level will specify algorithms on the level of specific gates Performed on one, two or more qubits [7]. They are probabilistic algorithm [12]. The basic algorithms are introduced here.

I. Deutsch's Algorithm
II. Deutsch-Jozsa Algorithm
III. Simon's Algorithm
IV. Peter Shor's Factorizing Algorithm
V. Lov Grover's Database Search Algorithm

1.1 DEUTSCH'S ALGORITHM

The Deutsch algorithm is an elementary quantum algorithm which is proposed by David Deutsch in 1985 [3]. Even it is in little practical use. It creates the first examples of a quantum algorithm which is more efficient than classical algorithm.

1.1.1 Procedure of Deutsch Algorithm

In the Deutsch problem, a black box quantum computer known as an oracle is given, that implements the function$f:\{0,1\} \rightarrow \{0,1\}$. Now the condition$f(0) = \mathrm{f}(1)$ needs to be checked. It is equivalent to check $f(0) \oplus \mathrm{f}(1)$ (where$\oplus$ is addition of modulo 2). It is not concerned to find the value or outcome of *f(x)* itself. To find the answer classically, one needs to query for both *x*=0 and x=1, hence two queries are required [5]. Quantum mechanically this can be solved in just one query. The figure represents the circuit for Deutsch's Algorithm.

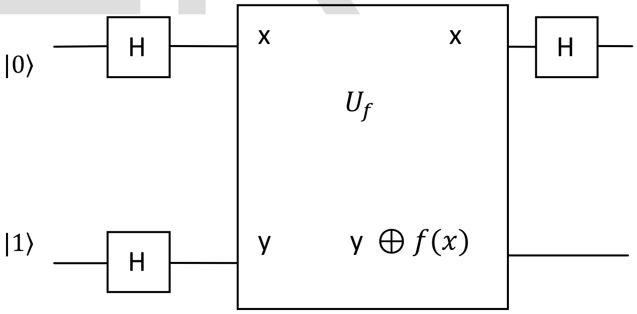

Fig 1: Circuit diagram of Deutsch's algorithm

Here, given a function$f:\{0,1\} \rightarrow \{0,1\}$, two qubits $|x,y\rangle$ are used and transferred them into $|x,y \oplus f(x)\rangle$. Two qubits are used to preserve reversibility, to keep the value of input x after the oracle performs. The second qubit y acts as a output register. Let *Uf* is the unitary transform which implements the function and maps$|x\rangle|y\rangle to|x\rangle|f(x) \oplus y\rangle$.

The process starts with the two qubits state $|0\rangle and|1\rangle$ and after this use Hadamard transform each qubit. This yield $\frac{1}{2}(|0\rangle + |1\rangle)(|0\rangle - |1\rangle)$.

After applying the function to the current state:

$$\frac{1}{2}(|0\rangle(|f(0)\oplus 0\rangle - |f(0)\oplus 1\rangle) + |1\rangle(|f(1)\oplus 0\rangle - |f(1)\oplus 1\rangle))$$

$$= \frac{1}{2}\left((-1)^{f(0)}|0\rangle(|0\rangle - |1\rangle) + (-1)^{f(1)}|1\rangle(|0\rangle - |1\rangle)\right)$$

$$= (-1)^{f(0)}\frac{1}{2}|1\rangle) + (-1)^{f(0)\oplus f(1)}|1\rangle(|0\rangle - |1\rangle).\rangle$$

The last bit is ignored and the global phase and therefore have the state by Hadamard transform it state:

$$\tfrac{1}{2}(|0\rangle + |0\rangle + (-1)^{f(0)\oplus f(1)}|0\rangle - |0\rangle - (-1)^{f(0)\oplus f(1)}|1\rangle)$$

$$= \tfrac{1}{2}((1 + (-1)^{f(0)\oplus f(1)}|0\rangle + (1 - (-1)^{f(0)\oplus f(1)}|1\rangle)\ .$$

If the result of measurement is a Zero, $^{f(0)\oplus f(1)=0}$. Therefore the function is constant and otherwise it is balanced. Here *Uf* is applied to 0 and 1 simultaneously. This is known as quantum parallelism. It provide square root improvement to query based problem [9].

1.1.2 Application

- It provides global property of solution space [11].
- It is the special case of the general Deutsch-Jozsa algorithm.

1.1.3 Advantage

- It provides exponential speedup over classical computer.
- Exponential improvement is possible for quantum computer.

1.1.4 Disadvantage

- It is only faster by a factor of 2.
- The time taken to solve the problem is same as classical computer.

1.2 DEUTSCH-JOZSA ALGORITHM

The **Deutsch–Jozsa algorithm** is discovering by David Deutsch and Richard Jozsa in 1992. It is one of the first examples of a quantum algorithm that is exponentially faster than any possible deterministic classical algorithm [1][2].

In this there is a black box quantum computer namely an oracle. It promised that the function is either constant (0 on all inputs or 1 on all inputs) or *balanced* (returns 1 for half of the input domain and 0 for the other half) the task then is to determine if *f* is constant or balanced by using the oracle.

1.2.1 Procedure of Deutsch-Jozsa Algorithm

In 1992, Deutsch and Jozsa produced a deterministic algorithm which was generalized to a function which takes *n* bits for its input. Further improvements to the Deutsch–Jozsa algorithm were made by Cleve resulting in an algorithm that is both deterministic and requires only a single query of *f*. This algorithm is still referred to as Deutsch–Jozsa algorithm in honour of the groundbreaking techniques they employed.

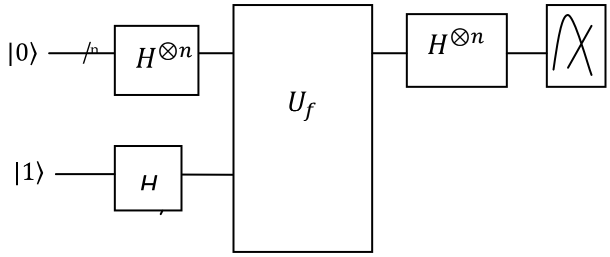

Fig 2: Circuit diagram of Deutsch-Jozsa algorithm

The algorithm begins with the n+1 bit state $|0\rangle^{\otimes n}|1\rangle$. That is, the first n bits are each in the state $|0\rangle$ and the final bit is $|1\rangle$.
A Hadamard transformation is applied to each bit to obtain the state

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle - |1\rangle)\ .$$

The function *f* implemented as quantum oracle. The oracle maps the state |x⟩ |y⟩to|x⟩|y $\oplus f(x)$ ⟩.

Applying the quantum oracle gives

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|f(x)\rangle - |1\oplus f(x)\rangle)\ .$$

For each x, $f(x)$ is either 0 or 1. A quick check of these two possibilities yields

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)\ .$$

At this point the last qubit may be ignored. Apply a Hadamard transformation to each qubit to obtain

$$\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\sum_{y=0}^{2^n-1}(-1)^{x\cdot y}|y\rangle = \frac{1}{2^n}\sum_{y=0}^{2^n-1}\left[\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{x\cdot y}\right]|y\rangle$$

Where

$x \cdot y = x_0y_0 \oplus x_1y_1 \oplus \cdots \oplus x_{n-1}y_{n-1}$ is the sum of the bitwise product.

Finally examine the probability of measuring $|0\rangle^{\otimes n}$,

$$\left|\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\right|^2$$

Which evaluates to 1 if *f(x)* is constant (constructive interference) and 0 if *f(x)* is balanced (destructive interference).

1.2.2 Application

- The algorithm taken originally two evaluations instead of one.
- It is based on quantum fourier transformation.

1.2.3 Advantage

- It is more efficient than any classical algorithm.
- Answer of the algorithm is always correct.
- It provide speed ratio 2^n:1.
- The algorithm was successful with a probability of one half.

1.2.4 Disadvantage

- It gives answer with a single evaluation.
- It takes two function evaluations instead of only one.

1.3 SIMON'S ALGORITHM

Simon's algorithm is one of the first quantum algorithms discovered which outperforms any known classical algorithm. The model of decision tree complexity conceived by Daniel Simon in 1994.

Simon's algorithm uses *o(n)* queries for black box and the best classical probabilistic algorithm necessarily needs at least $\Omega(2^{\otimes n/2})$ queries. It is also known that Simon's algorithm is optimal in the sense that *any* quantum algorithm to solve this problem requires $\Omega(n)$ queries.

The function $f{:}\{0,1\}^n \rightarrow \{0,1\}^n$, promised to satisfy the property that for some $s \in \{0,1\}^n$ for all $y, z \in \{0,1\}^n$, $f(y)=f(z)$, if and only if $y=z$ *or* $y \oplus z = s$.

1.3.1 Procedure of Simon's Algorithm

The set of *n*-bit strings is a z2 vector space . Given the preimage of *f* is either empty, or forms cosset with *n*-1 dimensions. Using quantum algorithms, can, with arbitrarily high probability determine the basis vectors spanning this *n*-1 subspace since *s* is a vector orthogonal to all of the basis vectors [8].

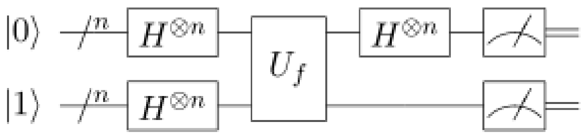

Fig 3: Circuit diagram of Simon's algorithm

The problem can be stated as a decision problem which goal is to decide whether or not there is a period that is whether f is 2 to 1 or 1 to 1. *Simon's problem* is an instance of an oracle problem which is classically hard, even for probabilistic algorithms, but tractable for quantum computers [4].

Classically the problem is hard because the probability to find two identical elements x and y after 2^(N/4) queries is less than 2^(−N/2). Simon's quantum solution is as the following

1. Start with a state vector$\langle H|0\rangle$) $\otimes^N |0\rangle \otimes^N$

2. Run the oracle once to make the state vector

$$2^{-\frac{N}{2}}\sum\nolimits_x |X\rangle|f(x)\rangle$$

3. Measure the second register; if the measurement outcome is $f(x_0)$, then the state vector of the first register will be

$$\frac{1}{\sqrt{2}}(|x_o\rangle + x_o \otimes p\rangle)$$

4. Applying a Hadamard gate to each of the N remaining qubits leads to

$$\frac{1}{2^{(N+1)}/_2}\sum\nolimits_y((-1)x_o.y + \left(-1^{\,(x_o\otimes p).y}\right)|y\rangle$$

$$=\frac{1}{2^{(N-1)}/_2}\sum\nolimits_{p.y-o}((-1)^{\,x_o.y}|y\rangle$$

Final measurement of the first register in computational basis, will give a value y which is such that y.p= 0 modulo 2.

Repeating this procedure in order to get N − 1 linearly independent vectors $y_1,\ldots,y_{N-1}$ p can be determined from the set of equations {yi. p = 0}. To this end there should be a procedure to query the oracle O(N) times.

1.4 PETER SHOR'S FACTORIZING ALGORITHM

This algorithm, first introduced by Peter Shor and used for integer factorization. On a quantum computer, to factor an integer N, the polynomial time taken in log$N$.

Peter Shor discovered the eponymous algorithm in 1994. It is very important because theoretically it can "break" the widely used public-key cryptography scheme RSA. It is based on the factoring of large numbers which is computationally infeasible for classical computers [10][14].

Shor algorithmcan divide into two parts.

a) Classical computer → a reduction of the factoring problem to a problem of order-finding.

b) Quantum computer→ An algorithm solving the order-finding problem.

1.4.1 Procedure of Shor's Factorizing Algorithm

Given n, find $2n^2 < q < 3n^2$ such that q is a product of small prime factors. We'll suppose $q = 2^1$ .Construct a quantum computer with $q^2 = 2^{21}$ qubits (plus additional qubits for 'workspace').The base states are denoted $|a, b\rangle = |a\rangle|b\rangle$ .Where a, b are binary vectors (i.e. vectors with entries 0,1) of length l. Equivalently, a and b (called *registers 1 and 2*) are integers < q written in binary.

At any time, the state of the system is given by

$$|\Psi\rangle = \sum_{a=0}^{q-1} \sum_{b=0}^{q-1} c_{a,b} |a, b\rangle \quad \text{where}$$

$c_{a,b} \in c$, $\Sigma_{a,b} |c_{a,b}|^2 = 1$ And $|c_{a,b}|^2$ is the probability that a measurement of the system will find the state to be $|a, b\rangle$.

Step 1

At initial state

$$|\Psi\rangle \,.......|(\Psi_m\rangle.$$

Step 2

Apply the randomly chosen value of x between 1 and n.

$$|a, x^a \, mod \; n\rangle.$$

to the state of the quantum computer.

$$q^{-1/2} \quad \sum_{a=0}^{q-1} |a, x^a \, mod \; n\rangle.$$

Step 3

Measure the second register only. observe the second register to be in a base state $|k\rangle$ where k is some power of x mod n (and all powers of x mod n are equally likely to be observed).

This measurement projects the state $|\Psi\rangle \in C^{q2}$ into the q-dimensional subspace spanned by all base states $|a, k\rangle$ for the fixed k whose value observed.

Thus the new state is

$$|\Psi\rangle \rightarrow \frac{1}{\sqrt{M}} \sum_{a \epsilon A} |a, k\rangle$$

Where A is the set of all $a < q$ such that x^a mod n is k and $M = |A|$. That is,

$A = \{a_0, a_0+r, a_0+2r, . . . , a_0+(M-1)r\}$

where $M \approx\approx \frac{q}{r} \gg 1$. thus

$$|\Psi\rangle = \frac{1}{\sqrt{M}} \sum_{d=0}^{M-1} |a_0 + dr, k\rangle.$$

Step 4

Apply the Discrete Fourier Transform Uq to the first register. This transforms the state from

$$\frac{1}{\sqrt{M}} \sum_{d=0}^{M-1} |a_0 + dr, k\rangle$$

to

$$= \sum_{c=0}^{q-1} e^{\frac{2\pi i c a_{0/q}}{\sqrt{qM}}} \left(\sum_{d=0}^{M-1} \zeta^d \right)|c, k\rangle$$

where $\zeta = e^{2\pi i cr/q}$.

Step 5

Measure register 1. observe register 1 to be in state$|c\rangle$ with probability

$$Pr(c) = \frac{1}{qM} \left| \sum_{d=0}^{M-1} \zeta^d \right|^2$$

where $\zeta = e^{2\pi i cr/q}$

if cr/q is not *very close* to an integer, then powers of ζ very nearly cancel out ('destructive interference') and such states$|c\rangle$ are extremely to be get.

Step 6

For the observed value of *c*, use a classical computer to find fractions *d/r* very close to *c/q*, hoping that this will give us the true order *r* of *x* mod *n*.

For this use the method of continued fractions, computing the convergent *d*1/*r*1 to *c/q* for which the denominator $r < n$. Noting that all the fractions$\frac{d_1}{r_1}, \frac{2d_1}{2r_1}, \frac{3d_1}{3r_1}$,.......... are close to *c/q*, it is reasonable to try small multiples of *r*1 as possible values of *r*.

$$r_1, 2r_1, 3r_1, \dots \dots, \lfloor \log_{(n)} 1+\in \quad \rfloor \, r_1$$

as possible values for *r*, checking whether x^r mod *n* gives 1 in each case, and repeating the experiment as often as necessary (*O*(1) times on average, compared with *O*(log log *n*) trials on average if multiples of *r*1 are not considered). Peter Shor's Algorithm is generalized to find the prime factors of an integer. Special quantum circuit design is proposed to find the divisors of a number. The lines required in a wiring diagram are proportional to n and the execution time is proportional to the square of n [5][6].

1.5 GROVER'S DATABASE SEARCH ALGORITHM

This is a quantum algorithm to find an unsorted database having *N* entries in *O* ($N^{1/2}$) time and using *O* (log*N*) storage space [4], this was invented by Lov Grover in 1996.

Conventionally, to search an unsorted database we requires a linear search, which is *O (N)* in time. Grover's algorithm, which takes *O* ($N^{1/2}$) time, is the quickest possible quantum algorithm to search an unsorted database. It provides "only" a quadratic speedup, in compare to other quantum algorithms. The quadratic speedup is considerable when *N* is large.

Grover's algorithm is probabilistic because of its ability to give the correct answer with high probability. The probability of failure can easily be decreased by repeating the algorithm [9].

The Grover's algorithm can be described as "inverting a function". If there is a function *y=f(x)* that can be evaluated on a quantum computer, described algorithm allows us to calculate *x* when *y* is given. Inverting a function is related to the searching of a database in the sense there can be a function which produces a particular value of *y* when *x* matches a desired entry in that database, and another value of *y* for any other values of *x*.Grover's algorithm can also be used for estimating the mean and median of a set of numbers, and for solving the Collision problem. It can also be used to solve NP-complete problems by performing exhaustive searches over the set of possible solutions [12].

1.5.1 Procedure of Grover's Algorithm

Let us consider an unsorted database having N entries. The *N*-dimensional state required for space *H*, which can be supplied by $\log_2 N$ qubits. Let's take the value for database entry1, 2, 3 ...N.Assume observation, *Ω*, acting on *H*, with *N* difference Eigen values which are known. Each of the eigenstates of *Ω* encodes one of the entries in the database, in a described manner. Eigenstates are denoted as $|1\rangle, |2\rangle, \dots., |N\rangle$ (using bra-ket notation) and the corresponding eigenvalues by $\{\lambda_{1,}\lambda_{2,\dots\dots\dots,}\lambda_N\}$.

A unitary operator is provided, U_{ω}, which acts as a subroutine that compares database entries with following some search criterion. The algorithm is not defined how this subroutine works, but it is *quantum* subroutine which works with superposition of states. Furthermore, it must act especially on one of the eigenstates, |*ω*>, which corresponds to the database entry matching the search criterion. To be precise, it is required U_ω to have the following effects: $(U_\omega|\omega\rangle) = (-|\omega\rangle), (U_\omega|x\rangle = |x\rangle), all\ x \neq \omega$

Our goal is to identify this eigenstate |ω>, or equivalently the eigenvalue ω, that U_ω acts especially upon. Two unitary operators are defined as follows: $U_\omega = I - 2|\omega\rangle\langle\omega|$ and$U_s = 2|s\rangle\langle s| - I$ after application of the two operators (U_ω and U_s), the amplitude of the searched-for element increases. And this is one Grover iteration *r*. $N=2^n$, *n* is number of qubits in blank (zero) state .

$$U_s|s\rangle = |s\rangle - \frac{2}{\sqrt{N}}|\omega\rangle \; And \; U_s\left(|s\rangle - \frac{2}{\sqrt{N}}|\omega\rangle\right) = \frac{N-4}{N}|s\rangle + \frac{2}{\sqrt{N}}|\omega\rangle$$

STEPS

The steps of Grover's algorithm:

1. Initialize the system to the state $|s\rangle = \frac{1}{\sqrt{N}} \sum_{x=1}^{N} |x\rangle$.

2. Perform the following "Grover iteration" *r(N)* times.

1. Apply the operator $U_\omega = I - 2|\omega\rangle\langle\omega|$.

2. Apply the operator $U_s = 2|s\rangle\langle s| - I$.

3. Perform the measurement Ω. The measurement result will be λ_ω with probability approaching 1 for N>>1. From λ_ω, ω may be obtained.

1.5.2 Application

- Finding a Witness for an NP Problem.
- Unstructured Database Search.

1.5.3 Advantage

- It is used for estimating the mean and median of a set of numbers.
- It is used for solving the Collision problem.
- It provides a quadratic speedup.
- It is probabilistic because it gives the correct answer with high probability.

1.5.4 Disadvantage

- In order to have high probability of success have to know the number t of solutions and t << N.

2 COMPARATIVE ANALYSES

NAME OF ALGORITHM	CONCEPT	APPLICATION	SPEEDUP
DEUTSCH'S ALGORITHM	It can complete task in one shot which take two shots in classical computer.	For global property solution.	It exhibits a two to one speedup in a certain computation.
DEUTSCH-JOZSA ALGORITHM	An exponenttial separation between classical deterministic and quantum algorithm.	For Fourier transform.	Even greater speedup with ratio 2^n: 1.
SIMON'S ALGORITHM	Exponential separation between probabilistic and quantum algorithm.	For decision tree.	
SHOR'S ALGORITHM	Quantum computer can efficiently factor numbers.	For order finding.	It provide the super polynomial speedup
GROVER'S ALGORITHM	It provides polynomial speedup over classical computer.	For searching database.	It provides quadratic speedup.

3 CHALLENGES

1. It must be scalable, it need a set of qubits that can be added to indefinitely.

2. The interaction between qubits must be controllable enough to make quantum logic gates.

3. There must be some readout capability.

4. It must be possible to move processing qubits accurately between specified locations.

4 CONCLUSIONS

By the help of quantum computing we increase the accessing speed of our computer for transferring the data, it's the basic step towards the quantum computing have been taken with demonstration and manipulation and its future scope is to fulfil all the challenges.

5 REFERENCES

[1] Cleve, A. Ekert, C. Macchiavello, and M. Mosca. "Quantum algorithms revisited", Proceedings of the Royal Society of London A454: 339–354(1998).

[2] David Deutsch and Richard Jozsa."Rapid solutions of problems by quantum computation". Proceedings of the Royal Society of London A439: 553(1992).

[3] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3, 1289-1305(Mar.2003).

[4] Grover L.K, A fast quantum mechanical algorithm for database search, Proceedings, 28th Annual ACM Symposium on the Theory of Computing, p. 212(May1996).

[5] Goong Chen, Stephen A. Fulling, Jeesen Chen, Generalization of Grover's Algorithm to Multiobject Search in Quantum Computing, Part I: Continuous Time and Discrete Time, quant-ph/0007123, Jul 2000.

[6] Hartmut Klauck, Quantum time-space tradeoffs for sorting, Proceedings of the thirty-fifth annual ACM symposium on Theory of computing, San Diego, CA, USA, Session 2A: 69 – 76(2003).

[7] John Preskill, Making Weirdness Work: Quantum Information and Computation.

[8] J. Eisert, M.M. Wolf, Quantum Computing.

[9] Lov K. Grover "A fast quantum mechanical algorithm for database search", Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing: 212–219(1996).

[10] Peter W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer.

[11] Sannella, M. J. Constraint Satisfaction and Debugging for Interactive User Interfaces. Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., University of Washington1994.

[12] Vidya Raj C and M. S. Shivakumar, Applying Quantum Algorithm to Speed Up the Solution of Hamiltonian Cycle Problems, IFIP International Federation for Information Processing, Springer Boston, Volume 228/2007:53-61

[13] Vishal Sahni, Quantum Computing, Tata McGraw Hill Education Private Limited New Delhi (2010)

[14] VlatkoVedral, Martin B. Plenio, Basics of Quantum Computation, Progress in quantum electronics, vol 22, (1998).

AUTHOR DETAILS

Milan Jain
M. Tech. Research Scholar
Computer Science & Engineering
Technocrats Institute of Technology, Bhopal
E-Mail:- jain.milan.jain@gmail.com

UNDER KIND GUIDANCE

Dr. Setu Kumar Chaturvedi
Professor& Head
Dept. of Computer Science & Engineering
Technocrats Institute of Technology, Bhopal
E-Mail:- setukchaturvedi@gmail.com